



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/449,021	11/24/1999	HELMUT EMMELMANN	373-02	5718

7590 06/06/2005

DERGOSITS & NOAH LLP
ATTN RICHARD A NEBB
FOUR EMBARCADERO CENTER
SUITE 1150
SAN FRANCISCO, CA 94111

EXAMINER

KENDALL, CHUCK O

ART UNIT	PAPER NUMBER
----------	--------------

2192

DATE MAILED: 06/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/449,021

Applicant(s)

EMMELMANN, HELMUT

Examiner

Chuck Kendall

Art Unit

2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 19 January 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1 - 8, 22 - 33, 41 - 43, 51 - 72, 74 - 96, 114 - 127 is/are pending in the application.
- 4a) Of the above claim(s) 9 - 21, 33 - 40, 44 - 50, 73, 97 - 113 is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1 - 8, 22 - 33, 41 - 43, 51 - 72, 74 - 96, 114 - 127 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is in response to the amendment filed on 01/18/05.
2. Claims 1 – 8, 22 – 33, 41- 43, 51 – 72, 74 – 96, 114 – 127 are pending in this Application.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 22 – 24, 26, 27, 30 – 33, 43, & 67 are rejected under 35 U.S.C. 102(e) as being anticipated by Truong USPN 6,151,609.

Regarding claim 22, a computer running an application to develop and maintain applications using a web browser comprising:

an editor operable with the web browser for inserting, deleting, and modifying component document templates (Truong, 11:17 – 20); and

a document generator for processing document templates, executing components and for generating documents from the document templates that are understandable by the web browser (Truong, 10:45-50).

Regarding claim 23, a computer as in claim 22, wherein the editor operates functional applications in an edit mode permitting editing directly in the web browser (Truong 2:1 – 5 & 10:45 – 50).

Regarding claim 24, a computer as in claim 23 wherein at least one of the components contains instructions and can react on subsequent document requests containing user responses by executing selected instructions (Truong, 10:45 – 50).

Art Unit: 2192

Regarding claim 26, Truong anticipates a system to modify documents on a server in a data network which couples said server computer to a client computer, the server computer comprising:

- a document store (3:27 – 30);

- a first program (3:32, see parser program) including instructions for transforming a first document retrieved from the document store (3:30 – 35, see editor input form) into a second document having features which permit editing of the first document such that at least a part of the second document appears and functions similar to the first document and (3:35 – 38, see editor selection form);

- a second software program including instructions to receive information from the client computer and instructions to modify documents stored in the document (3:35 – 38, see parser of client).

Regarding claim 27, the system of claim 26 wherein the first document includes at least one component being executed by the first software program (8: 13 – 15).

Regarding claim 30, the system of claim 26, wherein the features are scripts (Truong, 7:1-5).

Regarding claim 31, the system of claim 30, wherein the scripts encapsulate information from the first document (Truong, 9: 15 – 20).

Regarding claim 32, the system as in claim 26, wherein the features incorporate information regarding the first document into the second document (Truong, 9:10-15).

Regarding claim 33, a system as in claim 32, wherein the information incorporated into the second document is used on the client computer in order to send change requests for the first document to the server (Truong, 7:10-30).

Regarding claim 43, a system as in claim 26, additionally comprising at least one script for automatic download to the client that works in cooperation with the second document to permit editing of the first document (3:30 – 40, see parser and editor selection form, also see 9: 18, for JavaScript).

Regarding claim 67, see reasoning in claim (Truong, 8:39 – 50).

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1 – 8, 28, 29, 41, 42, 51 – 66, 68 – 96, 114 – 127 are rejected under 35 U.S.C. 103(a) as being unpatentable over by Truong USPN 6,151,609, in view of D'Arlach et al. USPN 6,026,433.

Regarding claims 1, 90, and 125, Truong discloses a software development system for applications that run on a data network which couples a server computer and a client computer, wherein the client computer runs a browser program, and whereupon request by the browser, an application generates generated documents for display by the browser and responds to the request with the generated documents pages, comprising:

a page generator running one of the application being developed by generating the generated documents including additional editing features for interpretation by the browser program (Truong, 10:45-50). Truong doesn't explicitly disclose an editor capable of directly operating on the documents displayed by the browser thereby allowing the user to work on a functional application during development. However, D'Arlach does disclose in analogous art, creating or editing a working copy of a user's site with the option of publishing the updated modified page or creating a new user web site (5: 15 – 25). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Truong and D'Arlach because, it would enable a working copy or user web page to be modified dynamically.

Regarding Claim 2, Truong discloses all the claimed limitations as applied in claim 1 including a plurality of components (2:1 – 5, for components see text boxes, buttons) and editor provides features to insert, modify and delete a component (10:47-50). Truong doesn't explicitly disclose wherein developed applications comprising document templates or editing components on templates and executing components on page templates. However, D'Arlach does disclose analogous art the use of document templates for modifying and creating a working copy of a new user sites, as well as containing customizable components in the template (5: 25 – 45). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Truong with D'Arlach to implement the instant claimed invention because, using templates would enable users to create and maintain a Web site easily and efficiently (D'Arlach, 4: 60 – 63).

Regarding Claim 3, a software development system as claimed in claim 2, wherein at least one of the components reacts interactively on user input by executing instructions on the server (Truong, 2:1 – 5).

Regarding claim 4, D'Arlach further discloses a software development system as in claim 3, wherein at least one of the components contains at least one other component (D'Arlach, 4: 63 – 65, see attributes and properties).

Regarding claim 5, D'Arlach further discloses a software development system as in claim 3, wherein the set of components on pages generated from at least one page template can vary for different of said page template (D'Arlach FIGURE 5,506).

Regarding claims 6, Truong discloses, a software development system for use in a data network which couples a server computer to a client computer, wherein the client computer includes a first software program for generating a request for one or more documents from the server computer and for displaying documents (Truong, 1: 65-67, 2:1-10, & 17-30), and wherein the server computer includes a second software program for receiving and processing the request from the client computer (Truong fig3, 3b 160, 10:45-50) for generating and storing documents, and for transmitting documents

to the client computer in response to requests, the server computer further comprising (Truong, 5:60-65) data store (fig 3c, 128), a plurality of components residing in the data store, including components that react interactively on user input (Truong, 2:1-5) by executing instructions on the server. Truong doesn't explicitly disclose a plurality of document templates residing in the data store, at least one document template and a third program selecting a document template based on the request from the client computer and instructions for generating a document from the document template for transmission to the client computer. However, D'Arlach does disclose in an analogous art a template database as well as style templates (5: 33 – 35). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Truong with D'Arlach to implement the instant claimed invention because, the use of templates would make customizing and maintaining web pages more efficient.

Regarding claim 7, which cites similarly to claim 5, see reasoning above as previously discussed.

Regarding Claim 8, the development system of claim 6, wherein a component is nested within a component (4: 60 – 65, for nested within component see, templates, components within them templates and objects/properties or attributes within the component).

Regarding claim 28, Truong discloses all the claimed limitations as applied in claim 27 above. Truong doesn't explicitly disclose wherein the second document includes handles and choosing one the handles selects a component for an editing operation. However, D'Arlach does disclose in an analogous art, selecting and editing templates which include editable components and objects (3:33 – 45). Therefore, it would have been obvious to one of ordinary skill in the art at the time then invention was made to combine, Truong and D'Arlach, because it would enable more efficient customization of objects during editing.

Regarding claim 29, the system of claim 28, wherein at least one handle indicates the position of atleast one component contained in a first document and

said editing operation includes modifying the component, deleting, the component, displaying information regarding the component (Truong, 11:17-20).

Regarding claim 41, a software development system as in claim 1, the editor comprising a client part for execution on the client computer (Truong, 3:35 – 38, see editor selection form).

Regarding claim 42, a software development system as in claim 41, wherein the client part comprises instructions for execution during editing that are automatically downloaded from the server in a request prior to editing (Truong, 3:30 – 38, see editor input form and selection form).

Regarding claim 51, see reasoning in claim 6.

Regarding claim 52, the system of claim 51, wherein components include fourth program instructions including steps to generate browser code prior to transmission to the first software program in response to a request from the first software program, wherein the browser code can differ for multiple request for the same document template [Truong, 7:1-15].

Regarding claim 53, a system in claim 52 running on a data network, coupling a server computer and a client computer, the first program running on the client computer, the second program running on the server computer [Truong, 7:10-30].

Regarding claim 54, a system in claim 52 wherein second documents include HTML documents with embedded scripts [Truong, 7:1-5].

Regarding claim 55, see reasoning in claim 6.

Regarding claim 56, D'Arlach further discloses, the system of claim 51, further comprising a fifth software program used by the second software program while processing selected document requests, the fifth software program including fifth instructions for generating generated documents from document templates thereby calling fourth program instructions (D'Arlach, FIGURE 5, 506, note: although a fourth and fifth program isn't discloses Prior art shows equivalent function as cited limitation).

Regarding claim 57, the system of claim 56, wherein the generated document includes, if requested in edit mode, edit features for interpretation by the first software program (Truong, 3:30 – 38, see editor input form and edit selection form).

Regarding claim 58, the system of claim 56, further comprising instructions to allow the user to click on the generated document to select items to perform edit functions on (D'Arlach, 10: 15 – 25).

Regarding claim 59, see reasoning in claim 6.

Regarding claim 60, the software development system of Claim 59 running on a data network, which couples a server computer and a client computer, the document generator running on the server computer the editor at least partly running on the client computer (Truong, 7:57 - 8:5).

Regarding claim 61, the software development system of claim 60, further comprising fourth instructions for execution during document generation to collect edit information for use by the editor (Truong, 8:39 – 50).

Regarding claim 62, the software development system of claim 60, wherein the editor uses a web browser for displaying said view (Truong, FIG. 3A, 108)

Regarding claim 63, Truong discloses all the claimed limitations as applied in claim 60 above. Truong doesn't explicitly disclose being able to automatically repeat requesting the document generator to process the dynamic web document if required. D'Arlach does disclose in an analogous art being able to process more templates as required see (D'Arlach, FIGURE, 620). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Truong and D'Arlach because, being able to repeat or re-request content would enable the system to make changes as required by user.

Regarding claims 64, D'Arlach further discloses the software development system of Claim 59 further comprising a plurality of components including at least one component marked on said dynamic web document, and including instructions for use by the document generator to generate browser code (Truong, see 9: 15 – 20, for javascript).

Regarding claim 65, the software development system of claim 64, wherein the editor uses a web browser for displaying said view (Truong, 8:39-50).

Regarding claim 66, see reasoning in claim 2.

Regarding claim 68 see reasoning in claim 61.

Regarding claim 69, the software development system of claim 68, wherein the editor uses the edit Information to correctly modify the dynamic web document (3:35 – 38, see editor selection form).

Regarding claim 70, see reasoning in claim 64.

Regarding claim 71, the software development system of claim 59, wherein the editor uses a web browser for displaying said view (Truong, 8:39-50).

Regarding claim 72, the software development system of claim 71, wherein first instructions comprise seventh instructions for initiating a reload in the browser [Truong, 7:59-67].

Regarding claim 73, D'Arlach further discloses the system of claim 59 wherein the editor program further comprises eighth instructions to display information on at least one element of atleast one dynamic web content, that is replaced during document generation, without requesting the document generator to generate the document (D'Arlach, 10: 15 – 25)

Regarding claim 74, see reasoning in claim 6.

Regarding claim 75, D'Arlach further discloses the software development system as in claim 74, wherein edit function comprises adding a component, modifying a component, and deleting a component (D'Arlach, 10: 10 – 25, and 50 – 60).

Regarding claim 76, the software development system as in claim 74, wherein tag syntax is used to denote atleast one component on at least one document templates, whereby the tag name identifies the component kind (Truong, 6:57 – 63).

Regarding claim 77, the software development system of claim 74, wherein running on a data network, which couples a server computer and a client computer, the document generator running on the server computer the editor running, at least partly, on the client computer (Truong, 3:35 – 38, see editor selection form);

Regarding claim 78, the software development system as in claim 74, wherein at least one component, that can react interactively on subsequent document requests, can upon selected document request for said document template be excluded from the generated document (D'Arlach, 10: 10 – 35).

Regarding claim 79, the software development system as in claim 78, further comprising third instructions to prevent excluded components from reacting on subsequent document requests (Truong, FIG. 3b, 164, and related text).

Regarding claim 80, a software development system as in claim 79, said third instructions comprising fourth instructions to, upon a first document request, store information in session memory on some of the components, that are present on the document generated based on the first request and fifth instructions to, upon subsequent document requests, only react on components that have been remembered in session memory there by avoiding tampering with excluded components on the side for the first program (D'Arlach, Examiner believes this to be equivalent to the storing of the customized template limitation as indicated in 5: 23 – 25).

Regarding claim 81, a software development system as in claim 74 wherein at least one first component contains sixth instructions to decide upon a request for said document template about exclusions of components nested inside the first component from the generated document (D'Arlach, 4: 60 – 65, Examiner interprets this limitation to be the customizable template limitation, which would include and exclude certain editable components).

Regarding claim 82, a software development system as in claim 74 the editor able to provide an editable view taking the varying set of components into account (Truong, 3:30 – 35, see editor input form and selection form).

Regarding claim 83, D'Arlach further discloses a software development system as in claim 74, the system able to provide an editable view that includes and excludes selected components on different requests for said document template similarly as the end user's view of the document template (D'Arlach, 4: 60 – 65, Examiner interprets this limitation to be the customizable template limitation, which would include and exclude certain editable components).

Regarding claim 84, see claim 4 for reasoning.

Regarding claim 85, D'Arlach further discloses the software development system as in claim 74, wherein multiple instances of a third component denoted on the

Art Unit: 2192

document template can be included in at least one of the generated document (D'Arlach, 6: 35 – 45).

Regarding claim 86, development system as in claim 85, further comprising seventh instructions to assign a unique identifier to each component instance, whereby the third component includes eighth instruction to qualify names generated into the browser code with the unique identifier (Truong, 8:20-35).

Regarding claim 87, D'Arlach further discloses a software development system as in claim 74, wherein at least one fourth component contains ninth instructions to decide upon a request about how many instances of components nested inside the fourth component are included in the document generated from said document template (D'Arlach, 4: 40 – 35 see request for specified action which Examiner interprets to be equivalent to number of instances of components nested, since the templates already a default set of components and a request from the client would involve altering, adding and deleting those components to customize it to the clients needs).

Regarding claim 88, see reasoning in claim 85.

Regarding claim 89, D'Arlach further discloses a software development system as in claim 74, wherein at least one sixth component includes tenth instructions to display the sixth component, the tenth instruction being used to generate the browser code for displaying the sixth component during editing as well as during normal use of the component (FIGURE 6, 604 also see related text, Examiner interprets this limitation to be equivalent function as components in D'Arlach's figure).

Regarding claim 90, an editor for use with a web browser, the editor allowing the user to edit atleast one document displayed by the browser, (Truong, 1: 65 - 67,2:1 - 10, & 17 – 30, also see 8:20 – 25);

wherein scripts contained in said document staying functional (Truong, 7:1-5), the editor comprising a first software program for execution within the browser and for processing selected clicks on the view of said document displayed in the browser by initiating editing functions [Truong, 10:45 – 50].

Regarding claim 91, the editor as in claim 90 using at least two windows, a first browser window displaying said document and a second window for displaying information on an element contained in said document (Truong, 6:55-65).

Regarding claim 92, the editor in claim 90 further comprising a second software program for modifying said, documents in cooperation with the first software program [10:45-50].

Regarding claim 93, the editor as in claim 92 further comprising a third program for transforming an original document into the document, the browser displaying the document as said view looking similar to the original document and interpreting editing features contained in the document [Truong, 7:20 – 30].

Regarding claim 94, editor in claim 93 wherein said original document is a dynamic document having components denoted thereon, the third software program further comprising instructions for generating browser code in cooperation with selected instructions contained in the components (Truong, 2:1 – 10).

Regarding claim 95, the editor as in claim 94 wherein the browser together with the first software program is running on a client computer connected to a server computer via a data network, wherein the second and the third software program run on the server computer (Truong, 1: 65-67, 2:1-10, & 17-30).

Regarding claim 96, the editor in claim 90 wherein links contained in said document stay functional allowing the user to browse and edit at the same time (8:39-45).

Regarding claim 114, a system for displaying dynamically generated documents in a data network coupling a server computer to a client computer, wherein the client computer has a first software program including first program instructions for generating a request to obtain at least one generated document from the server computer and for displaying the generated document, comprising:

a plurality of components on the server, at least one of the components including first features to cooperate with an editor in editing said component and second program instructions to generate browser code (Truong, 6:30 – 35);

third program instruction on the server for, generating generated documents for transfer to the client computer, thereby calling second program instructions of selected components (FIG.3A, see network editor page).

Regarding 115, the system of claim 114, wherein the first features include fourth program instructions for passing information to the editor (Truong, 10:55 – 59).

Regarding claim 116, system of claim 115 wherein at least part of said information is collected during execution of the components on the server (Truong, 7:60 – 67).

Regarding claim 117, system of claim 115 wherein said information is transmitted from the server to the client (7:60-8:5).

Regarding claim 118, system of claim 115 wherein said information includes attributes of said component (figure 5).

Regarding claim 119, system of claim 14, wherein first features include fifth instructions that display additional editing features of the components during editing (10: 45-50).

Regarding claim 120, the system of claim 119, where said editing features include handles (D'Arlach, FIGURE 6, 602 also see related text).

Regarding claim 121, system of claim 114, wherein first features include an extension for use by the editor, said extensions for enabling editing of an attribute value of the components (Truong, 10: 45-50, see editing features).

Regarding claim 122, system of claim 121 wherein said extension enables display of a page for editing the components attribute values (Truong, 8:65 – 67, also see FIG.3A, 108).

Regarding claim 123, system of claim 114 wherein at least one component is denoted on atleast one document templates using tag syntax, whereby the tag name identifies a component kind (Truong, 6:57- 63).

Regarding claim 124, system of claim 114, containing at least one component wherein second program instructions are used to generate browser code for displaying the component during editing and during normal uses (8:65-67).

Regarding claim 126, method of claim 125, wherein the running the step and the displaying step are repeated after applying a modification function (10: 45-50, see editing features also see FIG. 3A, 116).

Regarding claim 127 method of claim 125 further comprising collecting edit information for use by the identifying step (Truong, 8:45 – 52).

7. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over by Truong USPN 6,151,609, in view of D'Arlach et al. USPN 6,026,433 and further in view of Popp et al. USPN 6,651,108.

Regarding claim 25, Truong as modified by D'Arlach discloses all the claimed limitations as applied in claim 24 above. The combination of Truong and D'Arlach doesn't explicitly disclose a store of component classes, each component class implementing one component kind wherein the document generator works upon a document request using component classes. However, Popp does disclose in analogous art using HTML templates and classes to generate HTML pages (3: 15 – 40), in a similar configuration. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Truong, and D'Arlach with Popp because, the use of classes in an OO environment is a general practice and has been for years and would enable OO functionality and processing during Web development, hence making the web pages more compatible and user friendly.

Response to Arguments

Applicant's arguments filed 01/18/05 have been fully considered but they are not persuasive.

Argument (1), In Applicant's response regarding claim 1, Applicant argues that Neither Truong or D'Arlach teaches being able to "edit sever-side dynamic web applications".

Responding (1), Applicant's does not claim 1, In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., edit sever-side dynamic web applications") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Applicant's claims in claim 1 recites " an editor directly operating on pages displayed in the browser", and the Applicant's preamble discloses " ...data network which couples a server computer and a client, wherein the **client runs a browser**...". As claimed and as understood by Examiner, the editor is part of the browser and moreover claim does not claim editor being on the server side, hence Applicant's argument is moot for claiming an undisclosed merit of distinction.

Regarding claims 26, 27, 30 – 33, 43, and 67 Applicant argues that Truong does not teach a real time editor that does not permit real time editing. Applicant again is arguing for an unclaimed merit of distinction, therefore Applicant's argument is moot.

And regarding claim 26, Applicant argues that Truong only edits file list and not the file content. Examiner disagrees. In FIG. 4, Truong shows being able to select file names for editing and additionally also shows an icon for editing the file (file content).

Regarding claims 27,30 – 33, and 43, Applicant argues that Truong does not disclose the editor input form, which is a browser being built in HTML, Applicant again is arguing for an unclaimed merit of distinction, therefore Applicant's argument is moot.

Regarding claims 31, Applicant argues that claim requires " that the scripts transport information encapsulated in the first document to the browser". Applicant again is arguing for an unclaimed merit of distinction, therefore Applicant's argument is moot.

Regarding claim 32, Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable

invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

Regarding claim 33, see Truong 17:13 – 24, also refer back to previously cited section 7:10-30, which shows requesting web content from browser.

Regarding claim 43, with regards to HTML, Applicant again is arguing for an unclaimed merit of distinction, therefore Applicant's argument is moot.

Regarding claims 67,69,71 and 72, which depend on claims 59, have been amended and hence are new grounds of rejection, therefore Applicants argument is moot.

Regarding Argument in claims 1 – 8, 22 – 24, 28, 29, 41,42, 51 – 66, 68, 70, 73 – 96, and 114 – 127. Applicant argues that D'Arlach does not edit dynamic web application. Examiner disagrees. As previously disclosed D'arlach in 5:15 – 21, discloses editing a working copy of the template, which is the user's site, through forms displayed by the browser. Examiner interprets this as being able to edit dynamically. And moreover Applicant's claims do not exclude the use of Databases and Templates, in performing this task.

Regarding claims 3 – 5, Applicant does not disclose HTML elements interpreted by the browser being present on the server. Applicant again is arguing for an unclaimed merit of distinction, therefore Applicant's argument is moot.

Regarding claim 41, " the editor comprising: a client part for execution on the client computer", see 6: 32 – 35, for "Processor 34, interprets and executes instructions that have been fetched or retrieved from client library".

Regarding claim 42, which is a new ground of rejection, therefore argument is moot.

Regarding Applicant's argument in claims 6 - 8, Applicant asserts that in D'Arlach the templates do not have software components. Examiner disagrees, in D'Arlach 4:62 – 64, the templates are customizable and contain objects and elements which can be customized. Whether or not they are associated with or stored in the database should not matter since Applicant's claims does not exclude the Templates being associated with a database. And with regards to

Art Unit: 2192

objects and components not being equivalent, Examiner disagrees. Components are objects, and unless Applicant specifies in the claims a distinct feature which separates Applicant's component from the object discussed in D'Arlach then it will be interpreted that way. For example a graphical icon is both a component or an object, such as a text box or button. However a java bean is distinct.

Regarding claim 22, which is still being rejected by Truong using previously cited portions.

Regarding claim 11, Applicant argues Truong describes his text to HTML text which can include drop down list boxes, radio buttons, fill in text boxes, (6:60 – 65, Examiner again interprets this to be components.

Regarding claim 28 – 29, handles are inherent in graphical fill in textboxes, which are used to increase size, on all corners of the box with a mouse, unless Applicant defines his handles to be something else in claims, Examiner interprets his handles to be as noted.

Regarding claims 51 – 73, has been amended and hence is moot in view of new grounds of rejection.

Regarding claims 76, Applicant argues that HTML tag is not a component as claimed in Applicant's limitations. Examiner would like to indicate to Applicant that Applicant has merely claimed in his limitations "...wherein tag syntax is used to denote at least one component". Applicant argues that HTML tags in Truong (prior art) are not components, Applicant does not claim his tag being a component and hence Applicant's claim is moot in view of an unclaimed merit of distinction.

Regarding claim 78 – 81, has been amended and hence is moot in view of new grounds of rejection.

Regarding claim 83, Applicant is simply rehashing arguments, which have been previously discussed. See response above.

Regarding claims 84, 85 has been amended and hence is moot in view of new grounds of rejection.

Regarding claims 86, refer to previous rejection.

Regarding claims 87 and 88, new grounds of rejection.

Regarding claims 90, Applicant argues on page 29, of response that Truong does not teach being able to edit text and send back to the browser to execute the scripts. Examiner disagrees Truong in 8:20 – 25, Truong shows edited being interpreted by browser.

Regarding claim 91, Applicant argues that Truong does not talk about web browser windows, Examiner disagrees. Refer to drawing FIG. 3A, shows 102, executing at the web browser, which is a display and also 108, displaying network editor page, which is another display.

Regarding claims 92 – 94, new grounds of rejection.

Regarding claims 114, see claims above.

Regarding claims 115 – 125, Applicant states claims are dependent on claim 114 and asserts arguments on the same basis. And since claims 114 has been discussed above other arguments regarding claims 115 – 125, stemming from claim 114 have also been overcome.

Regarding claim 115, Applicant argues that prior art does not claim for instructions to be part of the components for execution on the server computer. Claims 15, however doesn't claim this. Applicant is arguing for an unclaimed merit of distinction hence Applicant's argument is moot.

Regarding claim 116, Applicant argues that the contents are not executed on the server, Examiner disagrees see claim 116, above which shows the remote editor which is loaded on the server.

Regarding claim 118, Applicants claims does not exclude editing features being indicated inside the browser, therefore Applicants argument is moot.

Regarding claim 120, Examiner has interpreted handles, as recited above. Applicant does not provide any further limitations to further distinguish his handles as claimed.

Regarding claim 121, Examiner has interpreted extension, as recited above. Applicant does not provide any further limitations to further distinguish his

limitation of extension from prior art, Examiner interprets the editing features in 10: 45-50, to be Applicant's limitations.

Regarding claim 122, Examiner believes that Prior art shows a page for editing component attributes see (FIG.3A, 108).

Regarding claim 123, Argument has already been discussed above.

Regarding claim 125, running an Application during editing see reasoning above in argument in claim 1.

Regarding claim 126, repeating the running step see (FIG. 3A, 116), loops back into process to denote repetition.

Regarding claim 127, Examiner interprets identifying from Truong to be identical to Applicant's identifying as claimed.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.


Art Unit: 2192

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chuck Kendall whose telephone number is 571-272-3698. The examiner can normally be reached on 10:00 am - 6:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

ck


WEI Y. ZHEN
PRIMARY EXAMINER

WEI Y. ZHEN
PRIMARY EXAMINER